Die vermeintliche Robustheit von Peer-to-Peer-Netzen

Jochen Dinger
Universität Karlsruhe (TH), Institut für Telematik
Email: dinger@tm.uni-karlsruhe.de

Hannes Hartenstein
Universität Karlsruhe (TH), Institut für Telematik
und Universitätsrechenzentrum
Email: hartenstein@rz.uni-karlsruhe.de

Abstract: Peer-to-Peer- (P2P-) Netze bilden logische bzw. virtuelle Netze auf Basis existierender Netze wie dem Internet. P2P-Netze zeichnen sich insbesondere durch ihre dezentrale Struktur und selbstorganisierende Mechanismen aus, welche im Gegensatz zu Client-Server Architekturen die Last auf alle teilnehmenden Knoten gleichmäßig verteilen. Durch die dezentrale Struktur in Kombination mit selbstorganisierenden Mechanismen ist es daher möglich, hoch skalierbare P2P-Netze zu konstruieren.

Zudem erhöht die Verteilung auf viele Knoten die Robustheit des Netzes gegenüber Ausfällen einzelner Knoten. P2P-Netze sind meist so gestaltet, dass beim Ausfall einzelner Knoten deren Aufgaben von einem anderen wahrgenommen werden. Dennoch sind P2P-Netze nicht per se robust, da neben dem vereinzelten Ausfall von Knoten auch gezielte Attacken möglich sind. Dieser Beitrag gibt einen Überblick über Attacken auf P2P-Netze. Außerdem wird die so genannte Sybil Attacke vertieft diskutiert und ihre Auswirkungen auf die Robustheit aufgezeigt. Abschließend wird ein neuer Ansatz präsentiert, um die Wahrscheinlichkeit einer solchen Attacke zu verringern.

1 Einleitung

Peer-to-Peer- (P2P-) Netzwerke erlangten in den letzten Jahren sowohl in der Praxis als auch der Wissenschaft eine hohe Aufmerksamkeit. Dienste zum Dateiaustausch (engl. File-Sharing) haben hierbei sicher zur großen Verbreitung beigetragen. Dennoch gibt es auch weitere praxisrelevante Einsatzgebiete wie IP-Telefonie, in denen P2P-Technolgie Verwendung findet. Im wissenschaftlichen Umfeld befasst sich ein Großteil der Diskussion mit so genannten verteilten Hashtabellen (engl. distributed hash tables - DHT), wie beispielsweise Chord [SMK+01]. Eine Übersicht findet sich unter anderem in [SW05].

P2P-Netze sind im Allgemeinen Overlaynetzwerke, das heißt oberhalb eines existierenden Netzes wie dem Internet wird ein (spezialisiertes) logisches Netz aufgebaut. Der Aufbau solcher Netze erfolgt möglichst ohne zentrale Koordinierungsinstanz, sondern dezentral durch selbstorganisierende Mechanismen. P2P-Netze sind durch die Fähigkeit zur Selbstorganisation 'selbstskalierend', da neu hinzukommende Knoten auch Aufgaben innerhalb des Netzes übernehmen und somit die Last auf alle Knoten verteilt wird. P2P-Netze nutzen meist ein eigenes Adressierungsschema, das den Knoten global eindeutige Kennungen

zuweist. (Die Knotenkennungen werden im Folgenden auch als Knoten-*ID* bezeichnet.) Insbesondere die DHT-basierten P2P-Netze nutzen diese Knoten-*IDs*, um ihre Struktur aufzubauen.

Die Robustheit eines P2P-Netzes gegenüber dem Ausfall einzelner Knoten ist in der Regel gegeben, da grundsätzlich alle Knoten gleich sind und daher jede Funktion von jedem Knoten erbracht werden kann sowie durch Replikation von Daten und durch alternative Routen ein Ausfall kompensiert werden kann. Betrachtet man jedoch gezielte Attacken von böswilligen Knoten, reduziert sich die Robustheit von P2P-Netze deutlich.

In Abschnitt 2 wird eine Zusammenfassung existierender P2P-bezogener Attacken gegeben. Abschnitt 3 geht auf die Sybil Attacke sowie die damit einhergehende Problematik der Zuweisung von Knotenkennungen näher ein. In Abschnitt 4 wird ein neuer Lösungsvorschlag mit dem Name 'Self-Registration' vorgestellt. Abschnitt 5 schließt den Beitrag mit einer Zusammenfassung und einem Ausblick ab.

2 Peer-to-Peer-spezifische Attacken im Überblick

Eine detaillierte Bedrohungsanalyse muss für jedes P2P-Netzwerk gesondert erfolgen, dennoch gibt es einige Attacken, die beim Design jedes P2P-Netzwerks berücksichtigt werden sollten. Sit und Morris [SM02] sowie Castro et al. [CDG+02] zeigten in ihren Arbeiten grundsätzliche Herausforderungen auf, welchen sich ein P2P-Netz stellen muss.

- **Sybil Attacke** – Die Sybil Attacke wurde von Douceur in [Dou02] erstmals erwähnt. Von einer Sybil Attacke spricht man, wenn es einem Angreifer gelingt, sich viele Identitäten zu beschaffen. Diese Identitäten kann ein Angreifer dann wiederum nutzen, um viele Knoten des P2P-Netzes zu kontrollieren und somit das Netz zu dominieren. Eine detaillierte Betrachtung folgt in Abschnitt 3.
- **Eclipse Attacke** – Im Falle einer so genannten Eclipse Attacke versucht ein Angreifer bzw. eine Gruppe von Angreifern, einen 'Opfer-Knoten auszublenden'. Der bzw. die angreifenden Knoten versuchen hierzu relevante Einträge in der Routing-Tabelle des Opfers so zu belegen, dass sämtliche Kommunikation des Opfers über einen der Angreifer stattfindet. Die Eclipse Attacke ist auch in 'Sybil-sicheren' P2P-Netzen möglich ist, da unter Umständen hierzu ein Knoten bzw. eine kleine Anzahl von Knoten ausreicht und diese nur passend platziert werden müssen. Diese Attacke richtet sich daher gezielt gegen einzelne Teilnehmer des Netzes während sich eine Sybil Attacke gegen das gesamte Netz richtet. In [SNDW06] wird die Eclipse Attacke detailliert untersucht und eine mögliche Abwehrmaßnahme vorgestellt. Hierbei zeigte sich, dass Optimierungsstrategien, die die Heterogenität des Netzes berücksichtigen, wie sie beispielsweise bei GIA [CRB+03] eingesetzt werden, insbesondere von einer Eclipse Attacke gefährdet sind.
- **Byzantine Join Attacke** – In DHT-basierten P2P-Netzen sind Knoten aufgrund ihrer Knoten-*ID* immer für einen Teil des *ID*-Raumes und daher auch für gewisse Daten zuständig. Werden diese Knoten-*IDs* von einem Angreifer nun gezielt aus-

gesucht, können bestimmte Daten unterdrückt werden. In diesem Fall spricht man von einer Byzantine Join Attacke. In [FSY05] wird eine Variante von Chord vorgestellt mit dem Namen S-Chord vorgestellt, welche diese Attacke unter gewissen Annahmen verhindert.

- **Routing Attacke** – Das Routing innerhalb eines P2P-Netz findet meist über mehrere Knoten statt, so dass durch böswilliges fehlerhaftes Routing die Funktion des P2P-Netzes gestört werden kann. Ein für die Weiterleitung verantwortlicher Knoten kann hierzu beispielsweise ein Paket an den falschen oder gar keinen Empfänger weiterleiten [CDG+02]. P2P-spezifische Routing-Tabellen werden aufgrund der dezentralen Struktur mit Hilfe der Nachbarknoten erstellt und sind somit auch Ziel von Attacken. Ein böswilliger Knoten kann versuchen, die Routing-Tabellen seiner Nachbarn durch falsche Aktualisierungen so zu verfälschen, dass Pakete falsch geleitet werden oder zu nicht existenten Empfängern geschickt werden.
- **Churn Attacke** – Sit und Morris erwähnten in [SM02], dass ständiges Beitreten und Verlassen des Netzes aufgrund des Wartungsaufwandes zu Problemen führen kann. Wenn diese 'Wechselrate' (engl. Churn Rate) eine gewisse Größe überschreitet, kann weder ein konsistenter Datenbestand noch ein korrektes Routing garantiert werden. Rhea et al. untermauern in [RGRK04] diese Vermutung durch Zahlen. Es wird beispielsweise gezeigt, dass in einem Pastry [RD01] Netzwerk (Pastry ist ein DHT-basiertes P2P-Netz.) mit 1000 Knoten bei einer mittleren Teilnahmedauer eines Knotens von 23 Minuten und unter realistischen Netzwerkbedingungen, kein konsistenter Schlüssel-Lookup mehr durchgeführt werden kann.

Neben den oben genannten P2P-spezifischen Attacken sind auch die üblichen Attacken wie Fälschen von Nachrichten, Angriffe durch wiederholte Pakete etc. möglich, dies beinhaltet auch Denial-of-Service Attacken gegen einzelne Knoten, das heißt die Überlastung eines Knotens durch übermäßige Anfragen. Zudem kann ein Angreifer sein Verhalten verschleiern, indem er Anfragen inkonsistent beantwortet. Hierzu verhält er sich gegenüber einem Teil der Knoten korrekt, während er anderen Knoten falsche Antworten schickt.

3 Die Sybil Attacke

Die Sybil Attacke stellt eine elementare Attacke dar, da durch sie alle Replikationsmechanismen ausgehebelt werden können und die weiteren geschilderten Attacken wesentlicher leichter umzusetzen bzw. 'nicht nötig' sind.

3.1 Terminologie

Für eine detaillierte Diskussion ist es notwendig, die Begriffe 'Teilnehmer' und 'Knoten' genau zu definieren. Im Folgenden wird eine entsprechende Terminologie eingeführt.

- Ein Knoten n_i ist die 'atomare' Einheit innerhalb eines P2P-Netzes. Die Menge

aller Knoten wird als $N = \{n_1, ..., n_n\}$ bezeichnet mit $|N| = n$ Anzahl beteiligter Knoten.

- Knoten sind die atomaren Einheiten auf P2P-Ebene, aber nicht zwangsläufig auf der darüber liegenden 'Gemeinschaftsebene' (vgl. [CDH+05]). Die Instanz über der P2P-Ebene wird im Folgenden als Teilnehmer p_i bezeichnet. Ein Teilnehmer kann ein oder mehrere Knoten kontrollieren. Alle Teilnehmer zusammen bilden die Menge $P = \{p_1, ..., p_m\}$ mit $|P| = m$. Es werden nur aktive Teilnehmer berücksichtigt, daher gilt $n \geq m$.
- Die Knoten-*ID* des Knoten n_i wird mit id_i gekennzeichnet. Die Kennungen sind aus dem Raum K, das heißt $id_i \in K$, $\forall i$. Wenn eindeutige Knoten-*IDs* in einem P2P-Netz zugewiesen werden, gilt: $\forall n_i, n_j \in N \; mit \; i \neq j : id_i \neq id_j$.
- Die Teilnehmer spezifischen Teilmengen werden mit N_i bezeichnet. Vorausgesetzt jeder Knoten gehört zu genau einem Teilnehmer, gilt: $\forall N_i \cap N_j = \emptyset \; mit \; i \neq j$ und $N_1 \cup \ldots \cup N_m = N$. Neben P2P-spezifischen Kennungen, können Knoten auch so genannte externe Kennungen besitzen, wie beispielsweise eine IP-Adresse oder ein PGP Zertifikat. Eine solche externe *ID* wird mit eid_i gekennzeichnet, wobei ein Knoten auch mehrere externe *IDs* besitzen kann.
- Außerdem soll noch die Konstante a definiert werden, die die maximale Anzahl erlaubter Knoten pro Teilnehmer kennzeichnet. Im 'Normalfall' gilt daher $\forall p_i \in P : |N_i| \leq a$. Im Falle einer 'erfolgreichen' Sybil Attacke gilt: $\exists p_i \in P : \; |N_i| > a$.
- $hash(i)$ bezeichne eine starke Einweg-Hashfunktion, wie beispielsweise SHA-256.

3.2 Definition und Auswirkungen

Unter einer Sybil Attacke versteht man gemäß der Definition von Douceur [Dou02], wenn es einem Teilnehmer gelingt, mehrere Identitäten in einem System zu besitzen. Bezogen auf P2P-Netze bedeutet dies, ein Teilnehmer erlangt Kontrolle über viele Knoten bzw. betreibt viele Knoten innerhalb eines P2P-Netzes. Es ist einem Angreifer dadurch möglich ein Netz zu dominieren und somit zu beeinflussen.

Um eine Sybil Attacke effektiv zu verhindern, ist es notwendig, dass jeder Knoten überprüfen kann, ob es sich bei der Identität eines anderen Knoten, um eine korrekte oder gefälschte Identität handelt. Douceur zeigte in seiner Arbeit, dass allein eine organisatorisch zentralisierte Vergabe der virtuellen Identitäten einen absoluten Schutz vor einer Sybil Attacke bieten kann, wobei Douceur davon ausgeht dass ein Knoten ausreichende Ressourcen besitzt, aber nicht unendlich viele Ressourcen. Jegliche andere Lösung wie Crypto-Puzzles etc. scheitert daran, dass die Berechnung entweder zu einfach ist oder aufgrund der Komplexität nicht ständig stattfinden kann und somit eine direkte Überprüfung der Identitäten nicht mehr gewährleistet ist. Indirekte Überprüfung erlauben es aber Angreifern, Teilbäume aus Sybils zu konstruieren.

Um die Auswirkungen der Sybil Attacke abschätzen zu können, ist es wichtig sich nochmals bewusst zu machen auf welcher Tatsache die Robustheit von P2P-Netzen beruht. Die

Robustheit der Netze beruht auf der Redundanz von Daten und redundanten Routen. Daten werden hierzu möglichst effizient auf verschiedene Knoten im Netz verteilt. Jeder Knoten unterhält meist auch Verbindungen zu mehreren Nachbarn, so dass er beim Ausfall einer Verbindung auf eine andere zurückgreifen kann. Gelingt es einem Teilnehmer nun eine Sybil Attacke durchzuführen, kann er unter Umständen alle Knoten kontrollieren, die für die Speicherung eines Datums verantwortlich sind und somit dieses Datum unterdrücken. Besitzt ein böswilliger Teilnehmer genügend Knoten und leitet Pakete nicht mehr oder falsch weiter, ist es ihm möglich das Netz zu partitionieren, da die meisten Routen dann immer über seine Knoten (Sybils) laufen. Schlussendlich führt eine 'erfolgreiche' Sybil Attacke zum Verlust sämtlicher Redundanz und somit auch zum Verlust der Robustheit des P2P-Netzes.

Die Auswirkungen einer Sybil Attacke auf so genannte 'Content Distribution'-Netze wie BitTorrent sind sicher geringer einzustufen als im Falle von IP-Telefonie im Stile von Skype. Bei BitTorrent erbringen die Knoten eine Art Proxy Funktionalität und jedem Knoten ist mindestens ein 'vertrauenswürdiger' Knoten bekannt, daher ist es zwar möglich die schnelle Verteilung des Inhaltes zu verhindern, jedoch nicht den Inhalt an sich zu unterdrücken. Betrachtet man aber P2P-basierte IP-Telefonie, könnte dort ein erfolgreicher Angriff zur Partitionierung des Netzes führen und somit die Erreichbarkeit der Teilnehmer stark einschränken bzw. unmöglich machen. In wie weit das Skype Netz nun tatsächlich gefährdet ist, lässt sich allerdings nur schwer beurteilen, da zum einen eine Nutzer-*ID* notwendig ist, welche (zentral) von Skype ausgegeben wird und zum anderen sind nur wenige Details des Protokolls öffentlich bekannt (vgl. [BD06]). Eine dritte Anwendungskategorie stellen Anonymisierungsdienste wie Freenet [CMH+02] dar, welche es Teilnehmern erlauben Daten anonym im Netz bereitzustellen oder anonymisiert zu kommunizieren. Solche Anonymisierungsdienste sind darauf angewiesen, dass ein gewisser Anteil der Teilnehmer unabhängig ist, da die Routen ansonsten wieder nachvollziehbar sind und die Anonymität nicht mehr gewährleistet ist.

Bei der Gestaltung von Reputationssystemen spielt die Sybil Attacke auch eine entscheidende Rolle. Cheng und Friedman zeigen in ihrer Arbeit [CF05], ein Sybil-sicheres Reputationssystem kann nicht auf einer symmetrischen Reputationsfunktion beruhen. Im Falle einer symmetrischen Reputationsfunktion wird allen Knoten gleich viel Vertrauen entgegen gebracht und die Knoten können daher beliebig ausgetauscht werden. Einem Angreifer ist es dann wiederum möglich einen Teilgraph zu konstruieren, mit welchem er sein Reputationswert beliebig anpassen kann.

Wie viele Knoten ein Teilnehmer besitzen muss um einen Angriff 'erfolgreich' durchzuführen, hängt sowohl von der tatsächlichen Größe des P2P-Netzes, als auch von der vorhandenen Redundanz innerhalb des Netzes ab. Daher lassen sich keine allgemein gültigen Grenzwerte festlegen, aber wie auch Douceur schon zeigte gibt es solch einen Punkt immer, außer es gelingt, die Knoten pro Teilnehmer effektiv zu limitieren.

Neben P2P-Netzen sind auch andere Netze, wie Sensor und Ad-Hoc Netze von der Sybil Attacke bedroht (vgl. [NSSP04]). Die Gefahr ist allerdings aufgrund der physikalischen Einschränkungen solcher Netze wesentlich geringer.

3.3 Knoten-*ID* Vergabe

Die Limitierung der Knoten pro Teilnehmer geht einher mit der Vergabe von Knoten-*IDs*. Knoten-*IDs* können auf verschiedene Arten vergeben werden, wie im Folgenden noch ausgeführt wird. Um gegen Sybil Attacken gewappnet zu sein, muss die Vergabe verifizierbar und limitiert sein. Ein Verifikationsmechanismus sollte es jedem Teilnehmer des P2P-Netzes ermöglichen, zu überprüfen ob eine Knoten-*ID* korrekt oder gefälscht ist. Beispielsweise könnte dies durch eine allgemein bekannte Invariante oder kryptographische Signatur gewährleistet werden. Limitierung bedeutet, es darf jenem Teilnehmer nicht möglich sein mehr als die erlaubte Anzahl von Knoten respektive Knoten-*IDs* zu erhalten. Die Vergabe kann in folgende vier Klassen unterteilt werden:

1. Durch eine zentralisierte *ID*-Vergabe kann sichergestellt werden, dass ein Teilnehmer maximal die Kontrolle über eine bestimmte Anzahl von Knoten hat. Beispielsweise könnte eine Certification Authority (CA) eine solche Aufgabe wahrnehmen.
2. Eine *ID*-Vergabe kann anhand von vorhanden 'externen Kennungen' erfolgen. Diese externen Kennungen werden dann basierend auf einer jedem Teilnehmer bekannten Abbildung, wie beispielsweise einer sicheren Einweg-Hashfunktion, auf eine P2P-spezifische Knoten-*ID* abgebildet. Die Knoten-*IDs* können beispielsweise aus der IP-Adresse abgeleitet werden.
3. Jeder Knoten kann sich eine beliebige *ID* aussuchen, die unter Umständen global eindeutig ist.
4. Die *ID* kann auch durch eine Gruppe von teilnehmenden Knoten bestimmt werden.

	Kosten	Eintritts-barriere	Verifikation	Limitierung
Zentrale Zuteilung	-	-	+	+
Dezentrale Zuteilung basierend auf externer *ID*	+	+	+	**?**
'Freie' dezentrale Zuteilung	+/0	+	+	-
Gruppenbasierte dezentrale Zuteilung	+/0	+	+	-

Tabelle 1: Klassifikation der Knoten-*ID* Vergabe

Tabelle 1 zeigt eine Gegenüberstellung der verschiedenen Vergabeverfahren. Kosten bezieht sich hierbei auf die Kosten, welche durch die *ID*-Vergabe entstehen, wobei ein '+' nicht weniger, sondern höhere Kosten bedeutet. Die Erlangung einer Knoten-*ID* ist unter Umständen mit erheblichem Aufwand verbunden, wie beispielsweise im Falle einer CA, die eine Authentifizierung per Personalausweis verlangen könnte und somit würde die Eintrittsbarriere in das P2P-Netz wesentlich erhöht werden. Verifikation und Limitierung sind wie oben definiert. Die zentrale Vergabe von P2P-Knoten-*IDs* ist nicht ohne weiteres möglich, da solch eine zentrale Vergabestelle zunächst einmal eingerichtet

werden muss und der Betrieb Geld kostet. Außerdem steht der Gedanke im Widerspruch zu der völligen Dezentralität des P2P-Netzes. Die Ableitung der Knoten-*ID* von externen Kennungen, wie IP-Adresse, ist daher der vielversprechendste Ansatz, da in diesem Fall auf bekannte Merkmale zurückgegriffen wird und somit die Kosten und Eintrittsbarriere gering sind. Eine Verifikation kann beispielsweise durch die Invariante der Form $\forall n_i \in N : hash(eid_i) - id_i = 0$. Offensichtlich wird das Problem teilweise auf Organisationen, wie ICANN etc. verlagert, dies ist aber legitim, da diese externen Kennungen bereits vorhanden sind. Wie im vorderen Teil der Arbeit schon erwähnt, sind die freie Vergabe und Gruppen-basierte Vergabe nicht sicher genug, da eine effektive Limitierung laut Douceur nicht möglich ist. Zudem verursacht der Einsatz von Crypto-Puzzles oder ähnlichem teilweise nicht unerhebliche Rechenkosten.

3.4 Auswirkungen mehrerer Knoten-*IDs* pro Teilnehmer

Wenn es böswilligen Teilnehmern gelingt mehr als einen Knoten zu besitzen, hat dies immense Auswirkungen auf das Verhältnis zwischen gutartigen und böswilligen Knoten. Der Parameter a entspricht hierbei der Anzahl Knoten eines böswilligen Teilnehmers.

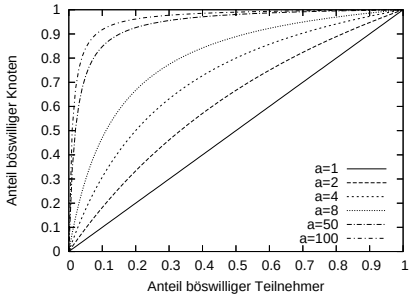

Abbildung 1: Verhältnis zwischen böswilligen Teilnehmern p_p und dem resultierenden Anteil böswilliger Knoten p_n

Das Verhältnis zwischen dem Anteil böswilliger Teilnehmer p_p und böswilliger Knoten p_n wird wie folgt berechnet, unter der Annahme dass gutartige Teilnehmer nur ein Knoten besitzen. (n_{mal} bezeichnet die Anzahl böswilliger Knoten):

$$p_n = \frac{n_{mal}}{n} = \frac{a * p_p}{(1 - p_p) + a * p_p} \tag{1}$$

Abbildung 1 zeigt, dass der tatsächliche Anteil böswilliger Knoten sehr schnell ansteigt, selbst für relativ kleine Werte von a. Nimmt man beispielsweise $a = 8$ und 2% böswilliger Teilnehmer an, resultiert dies in mehr als 14% böswilliger Knoten. Daher ist ein relativ geringer Anteil böswilliger Teilnehmer ausreichend, um eine Netz zu dominieren. Bei einer Bedrohungsanalyse ist es daher notwendig vom Anteil böswilliger Teilnehmer auszugehen und a korrekt abzuschätzen, um realistische Ergebnisse zu erhalten.

4 Sybil-Resistenz durch Self-Registration

Ausgehend von den verschiedenen *ID*-Vergabe Verfahren, die in Abschnitt 3.3 vorgestellt wurden, basiert unser Self-Registration Ansatz auf externen *IDs*. Wir nutzen hierzu die IP-Adresse des Teilnehmers. Die Ableitung sowie Verifikation der Knoten-*ID* ist durch eine sichere Einweg-Hashfunktion möglich. Beim Einsatz von Network Address Translation (NAT) oder IPv6 mit Privacy Extensions entsteht jedoch das Problem, dass im ersten Fall die Knotenanzahl bei NAT auf einen Knoten begrenzt wäre und im zweiten Fall hätte jeder Teilnehmer potentiell beliebig viele IP-Adressen. Zur Lösung dieser beiden Probleme, erlauben wir einerseits mehrere Knoten-*IDs* pro IP-Adresse und im Falle von IPv6 nutzen wir nur einen Teil der IP-Adresse. Unter der Annahme, dass heutzutage jeder Teilnehmer nur eine begrenzte Anzahl von gültigen öffentlichen IPv4-Adressen (gleichzeitig) besitzt, ist eine Limitierung der Knoten-*IDs* pro Teilnehmer gegeben. Wenn man im Falle von IPv6 nur die ersten 64-bit der IP-Adresse nutzt ist es zwar immer noch möglich, dass ein Teilnehmer mehrere IP-Adressen besitzt, aber aufgrund der Routing Eigenschaften ist es einem Teilnehmer nur möglich in den Besitz einer sehr begrenzten Anzahl von IPv6 Präfixen zu gelangen. Wir bezeichnen die IPv4 Adresse bzw. den IPv6 Präfix des Knoten n_i im Folgenden als $ipAddressPre_i$.

Kurz umrissen funktioniert unser Ansatz folgendermaßen: Knoten berechnen basierend auf ihrer IP-Adresse und Port eine Knoten-*ID* und registrieren diese dann im Netz selbst auf Basis des IP-Adressen Päfixes $ipAddressPre_i$. Die Registrierung erfolgt bei mehreren Knoten, da im P2P-Netz schon böswillige Teilnehmer vorhanden sein könnten, welche den Ansatz ansonsten sofort torpedieren könnten. Die Anzahl r von Registrierungsknoten ist hierbei frei wählbar. Dieser Ansatz ist zwar nicht absolut 'Sybil-sicher', aber wir evaluieren mit Hilfe dieses Ansatzes, welches Maß an Sybil-Resistenz gewährleistet werden kann und wie lange ein solches Netz einer Sybil Attacke widerstehen kann.

4.1 Algorithmus

Unser Self-Registration Ansatz basiert auf dem Chord Protokoll [SMK+01], sollte aber mit geringem Aufwand auf andere DHT-basierte P2P-Netze übertragbar sein. Der Ansatz verlangt die Anpassung der so genannten Join- und Stabilize-Prozesse in Chord. Der Join-Prozess wird wie folgt erweitert:

1. Ein neuer Knoten n_i berechnet zunächst durch eine Hashfunktion seine Chord *ID* id_i, wobei die id_i auf IP-Adresse und Port basiert.
   Somit gilt: $eid_i = ipAddress_i \oplus port_i$, $hash(eid_i) = id_i$.
2. Nachdem der Knoten n_i seine Kennung id_i berechnet hat, registriert der Knoten diese id_i bei den r zuständigen Registrierungsknoten. Diese Registrierungsknoten werden wie folgt berechnet: $regId_i^j = hash(j \oplus ipAddressPre_i)$ $(1 \leq j \leq r)$ (Das Symbol $\oplus$ bezeichnet die Verknüpfung von Strings.) Hierbei fließt nur die IP-Adresse (IPv4) bzw. im Falle von IPv6 der Präfix der IP-Adresse ein, so dass die Registrierung Teilnehmer basiert ist.

3. Abschließend versucht der Knoten n_i dem P2P-Netz beizutreten und informiert hierzu seinen Nachfolger im Chord-Netz. Mit Hilfe des sich wiederholenden Stabilize-Prozesses wird der Knoten dann ins Netz integriert.

Erreicht eine Registrierungsanfrage einen verantwortlichen Registrierungsknoten, prüft dieser zunächst ob die Anzahl registrierter Knoten für diese IP-Adresse bzw. den Präfix $ipAddressPre_i$ kleiner als a ist. Ist dies der Fall speichert der Registrierungsknoten die Knoten-*ID* id_i sowie die IP-Adresse und Port (eid_i). Jeder Knoten speichert daher eine Liste von IP-Adressen und für jede IP-Adresse $ipAddressPre_i$ eine Liste aktuell registrierter Knoten. Der Stabilize-Prozess wird wie folgt erweitert:

1. Trifft eine 'join'-Anfrage eines neuen Knotens n_i bei einem existierenden Knoten n_j ein, versucht der Knoten n_j zuerst die id_i mit Hilfe der Invariante $hash(eid_i) - id_i = 0$ zu verifizieren. Zudem überprüft der Knoten n_i, ob der Knoten n_j im Besitz der eid_i ist, durch eine Art 'Ping'-Nachricht.
2. Danach berechnet der Knoten n_j die verantwortlichen Registrierungsknoten $regId_i^j$ in der gleichen Weise wie sie n_i während des Join-Prozesses berechnet hatte. Abschließend konsultiert n_j die Registrierungsknoten und überprüft ob sich der Knoten dort ordnungsgemäß registriert hat.
3. Wenn die Anzahl positiver Antworten $\geq \lceil \frac{r}{2} \rceil$ ist, wird der neue Knoten n_i in das P2P-Netz integriert. Die Anzahl positiver Antworten wird mit k bezeichnet.

Nachdem ein neuer Knoten erfolgreich dem Netz beigetreten ist, nimmt dieser auch am Registrierungsprozess teil. Daher müssen unter Umständen auch (Registrierungs-)daten, gemäß des Chord-Protokolls transferiert werden. Verlässt ein Knoten das Netz sind analoge Datentransfers notwendig.

4.2 Evaluierung

Zur Evaluierung haben wir Chord sowie die in Abschnitt 4.1 beschriebenen Erweiterungen in der Simulationsumgebung J-Sim [Tya05] implementiert. Unser Simulationsaufbau war wie folgt: Teilnehmer bzw. ihre Knoten versuchen dem Netz beizutreten und verlassen dieses nach einer gewissen Zeit wieder. Böswillige Teilnehmer registrieren zunächst die erlaubte Anzahl a Knoten und versuchen dann weitere Knoten 'illegal' zu registrieren. Solche 'illegalen' Registrierungsversuche werden periodisch wiederholt. Zudem erlauben böswillige Knoten die Registrierung einer beliebigen Anzahl von Knoten, das heißt sie untergraben die Limitierung. Daher kann es böswilligen Teilnehmer gelingen, mehr als die erlaubte Anzahl von Knoten zu registrieren, wenn der Parameter a nicht korrekt abgeschätzt wurde und r nicht entsprechend angepasst.

Abbildung 2 zeigt einen Simulationslauf mit 2% böswilliger Teilnehmer ($p_p = 0.02$) sowie den Parametern $a = 2$ und $r = 5$. Der Graph zeigt, dass die Anzahl böswilliger Knoten ungefähr der Anzahl erwarteter böswilliger Knoten entspricht und nie signifikant abweicht. Somit war es den böswilligen Teilnehmern nicht möglich eine erfolgreiche Sybil Attacke zu starten. Während der Simulation tritt ein neuer Teilnehmer alle

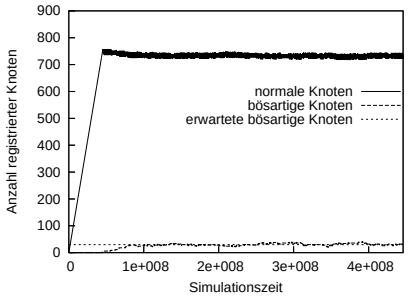

Abbildung 2: Simulationslauf einer durch 'Self-Registration' verhinderten Sybil Attacke

60.000 Zeiteinheiten (Simulationszeit) dem Netz bei und der Stabilize-Prozess wird alle 14.000 Zeiteinheiten ausgeführt. Nach $45 * 10^6$ Zeiteinheiten verlassen die Teilnehmer und ihre Knoten das Netz wieder. Ein neuer Teilnehmer ist mit der Wahrscheinlichkeit $(1 - p_p)$ gutartig bzw. mit Wahrscheinlichkeit p_p böswillig. Ein böswilliger Teilnehmer versucht dann nach seiner Erstellung so viele Knoten wie möglich zu registrieren. Die Anzahl erwarteter böswilliger Knoten wird wie folgt berechnet (vgl. Abschnitt 3.4): $p_n \approx 0.0392 \Rightarrow n_{malicious} = m * p_n \approx 30$ ($m = \frac{lifetime}{creationInterval} = \frac{45*10^6}{60{,}000} = 750$).

Für die weiteren Evaluierung wurde $a = 1$ gesetzt. Für $a > 1$ müssen die Ergebnisse mit den Ergebnissen aus Abschnitt 3.4 kombiniert werden, daher gilt auch $p_p = p_n$. Sind $k \geq \lceil \frac{r}{2} \rceil$ böswillige Knoten in den Registrierungsprozess involviert, kann es einem neuen böswilligen Knoten gelingen in das Netz integriert zu werden. Dies wird im Folgenden als 'illegale Registrierung' bezeichnet.

$$g(r, k, p_n) = (1 - p_n)^{r-k} * p_n^k * \binom{r}{k} \tag{2}$$

$$h(r, p_n) = \sum_{i=\lceil \frac{r}{2} \rceil}^{r} g(r, i, p_n) \tag{3}$$

Die Wahrscheinlichkeit, dass k Knoten eine Registrierung eines böswilligen Knoten bestätigen, berechnet sich anhand der Funktion $g(r, k, p_n)$. Die Wahrscheinlichkeit einer 'illegalen Registrierung' entspricht $h(r, p_n)$. Abbildung 3 zeigt die Wahrscheinlichkeiten für einen Replikationsfaktor von $r = 5$. Wird der Replikationsfaktor r erhöht, sinkt die Wahrscheinlichkeit einer 'illegalen Registrierung'. Abbildung 4 zeigt diesen Zusammenhang.

Die Ergebnisse zeigen, dass der Self-Registration Ansatz ein akzeptables Niveau der Sybil-Resistenz bietet, wenn die Parameter r und a passend gewählt sind. Die Wahrscheinlichkeit einer 'illegalen Registrierung' kann durch Erhöhung des Replikationsfaktors r deutlich verringert werden, was allerdings zusätzlichem Kommunikationsaufwand nach sich zieht. Weitere Details des Self-Registration Ansatzes finden sich in [DH06].

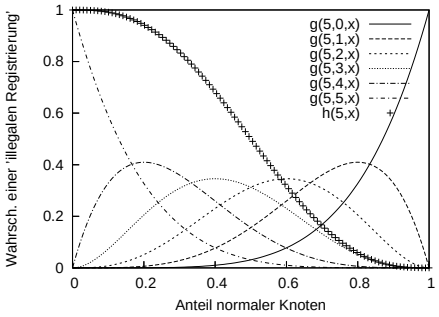

Abbildung 3: Wahrscheinlichkeit einer 'illegalen Registrierung' im Vergleich zum Anteil gutartiger Knoten

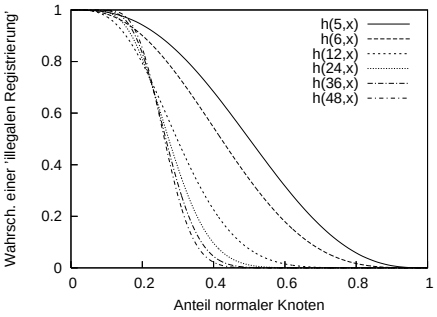

Abbildung 4: Verschiedene Replikationsfaktoren und die daraus resultierende Wahrscheinlichkeit einer 'illegalen Registrierung'

5 Zusammenfassung und Ausblick

Dieser Beitrag gibt einen Überblick über spezifische Attacken in P2P-Netzwerken und zeigt die Auswirkungen hinsichtlich der Robustheit solcher Netze auf. Die Sybil Attacke stellt hierbei eine sehr effektive Attacke dar, die sämtliche Robustheit eines P2P-Netzes zunichte machen kann.

Nach den ernüchternden Ergebnissen von Douceur [Dou02], die nahe legen, dass nur zentralisierte Lösungen zur Knoten-*ID* Vergabe Sybil Attacken verhindern können, haben wir analysiert, inwiefern sich bereits bestehende externe Kennungen für Sybil-resistente P2P-Netze nutzten lassen und mit dem 'Self-Registration' Mechanismus auch einen neuen Ansatz präsentiert. Unser Ansatz verspricht zwar keine vollständige Sicherheit, dennoch besteht ein Potential der Sybil-Resistenz, welches einer Analyse im probabilistischen Sinne zugänglich ist. Unserer Meinung nach ist dieser Ansatz der probabilistischen Sicherheit auch vielversprechender als die absolute Sicherheit, da sich absolute Sicherheit nur schwer bis gar nicht erreichen lässt. Ausgehend von den hier präsentierten Analysen, werden wir künftig die Sybil-Resistenz unseres und weiterer Ansätze mit Hilfe stochastischer Prozesse genauer untersuchen.